STORE COPY

MSA BASIC

1200 BAUD
CUTS FORMAT
(REB CUTER)

TABLE OF CONTENTS

SECTION		PAGE
1.0	Bibliography	1
2.0	Application Software References	2
3.0	Loading BASIC	3
3.1	Using Solos on the SOL-20 Terminal Computer	3
3.2	Using Cuter on the BYT-8, IMSAI, ALTAIR VectorGraphics	3
4.0	Initializing BASIC	4
4.1	Core Allocation	4
4.2	Line Length	5
4.3	Trigonometric Functions	5
4.4	Bytes of Core Available	5
5.0	BASIC Commands	6
6.0	BASIC Statements	7
7.0	BASIC Functions	8
8.0	Special Characters	9
9.0	Error Messages	10
	ADDENDUM	

1.0 BIBLIOGRAPHY

The documentation presented herein is not intended to be a tutorial for Basic but, rather, a reference of the features available under MSA BASIC. The user should already be familiar with the Basic language, and its syntactical structure. If, however, you are a beginner or need to refresh your abilities, please consult the books listed below before using MSA BASIC.

1. My Computer Likes Me When I Speak in Basic; by Bob Albrecht, published by Dymax, Menlo Park, California.
2. Basic Basic; by James S. Coan, published by Hayden Book Company, Rochelle Park, New Jersey.
3. Advanced Basic; by James S. Coan, published by Hayden Book Company, Rochelle Park, New Jersey.
4. Basic Programming Language - Programmed Learning Aid; by Brady and Richardson, published by Learning Systems Company, Homewood, Illinois.
5. Instant Basic; by Jerald R. Brown, published by Dymax, Menlo Park, California.
6. Basic Programming Language; by Allen M. Brady, Irwin-Dorsey, Ltd., Georgetown, Ohio.

NOTE: All of the above books are available from all retail Computer Stores.

2.0 APPLICATION SOFTWARE REFERENCES

1. Some Common Basic Programs; by Lon Poole and Mary Borchers, published by Adam Osborne, and Associates, Inc., Berkeley, California.
2. 101 Basic Computer Games; by David H. Ahl, published by Creative Computing, Morristown, New Jersey.
3. Basic Software Library, vol. I-VI; by R. W. Brown, published by Scientific Research Inst., Crofton, Maryland.

NOTE: All of the books listed above can be purchased from all retail Computer Stores.

3.0 LOADING MSA BASIC

3.1 USING SOLOS ON THE SOL-20 TERMINAL COMPUTER

A. Power up the SOL-20

B. Make sure that your system has at least 8-K of RAM from 0000H to 2000H.

C. Make sure that a tape recorder is connected to your SOL-20 and the cassette interface is functionally operational.

D. Type in XEQ BASIC

E. Hit the return key

F. Put tape in recorder

G. Hit play button on recorder

H. BASIC is self initializing after it has loaded - see initialization.

3.2 USING CUTER ON THE BYT-8, IMSAI, VECTORGRAPHICS, ALTAIR

A. Cuter must be operating from C000H to C800H

B. 8K of RAM must be present in your system from 0000H to 2000H.

C. Make sure that the tape recorder and cassette interface are connected and operational.

D. Power up system and type XEQ BASIC

E. Hit the return key

F. Put tape in recorder

G. Hit play button on recorder

H. BASIC is self initializing after it has loaded - see initialization.

4.0 INITIALIZING MSA BASIC

After BASIC has loaded, it will type on the output device:

MSA BASIC

CUTER VERSION	SOLOS VERSION
COPYRIGHT (C) 1977	COPYRIGHT (C) 1977
MICROCOMPUTER SOFTWARE	MICROCOMPUTER SOFTWARE
ASSOCIATES	ASSOCIATES

4.1 CORE ALLOCATION?

At this time, BASIC is asking you how much memory you want to allocate for its use.

1. To allocate all continuous memory starting at 0000H then hit return (automatic allocation mode).
2. To allocate only part of memory, type in the "decimal" number that corresponds to the upper limit you want BASIC to use, and then hit return, example: 49,151 for three 16K boards.

BASIC will then type:

4.2 LINE LENGTH?

BASIC is now asking how wide your terminal display is.

1. To use a 72 character wide terminal, hit return.
2. To use a 64 character wide terminal, type 64 then hit return.
3. To use any other "non-standard" terminal, type in the decimal number of characters that can be displayed on one line.

BASIC will then type:

4.3 TRIGONOMETRIC FUNCTIONS?

BASIC is now asking if you would like the standard trigonometric functions.

1. Type Y then hit return to retain all functions.
2. Type N then hit return to delete all trig functions.

BASIC will use approximately 200 bytes less if the trig functions are deleted.

BASIC will then type:

4.4 99999 BYTES OF CORE AVAILABLE

READY

Where "99999" equals the decimal amount of bytes available for program storage in the system.

BASIC is now loaded, initialized, and ready to use.

5.0 MSA BASIC COMMANDS

COMMANDS	DESCRIPTION	EXAMPLE
CLEAR	Clears string space	CLEAR 200
CONT	Continues program after C	CONT
LIST	Lists program on terminal	LIST 500
LOAD	Loads program from cassette	LOAD "A"
LOAD *	Loads array from cassette	LOAD *A
NEW	Erases program in memory	NEW
NULL	Sets nulls to terminal	NULL 3
RUN	Runs program in memory	RUN 1000
SAVE	Saves program on cassette	SAVE "A"
SAVE *	Saves array on cassette	SAVE *A

NOTE: C implies Control C.

6.0 MSA BASIC STATEMENTS

STATEMENTS	DESCRIPTION	EXAMPLE
DATA	Sets data elements in program	1000 DATA 1,2,3,4
DEF	User defined function	1000 DEF FNA(X)=X^2+X+1
DIM	Dimensions arrays	10 DIM A (5,5) , A$ (10,5)
END	End of program	9999 END
FOR	For/next looping	100 FOR X=1 to 100
GOTO	Unconditional branch	100 GOTO 200
GOSUB	Subroutine branch	100 GOSUB 200
IF...GOTO	Conditional branch	100 IF X=0 GOTO 200
IF...THEN	Conditional execution	100 IF X=0 THEN Y=1
INPUT	Terminal input	100 INPUT "DATA"; D
NEXT	For/next terminator	300 NEXT
ON...GOTO	Conditonal branch	100 ON X GOTO 10,20,30
ON...GOSUB	Conditional subroutine branch	100 ON X GOSUB 10,20,30
OUT	Direct terminal output	100 OUT 1,255
PEEK	Memory inspect	100 PEEK 1000
POKE	Memory change	100 POKE 1000,255
PRINT	Prints on terminal	100 PRINT X,Y,Z
READ	Reads from data statement	100 READ A,B,C
REM	Remark	100 REM THIS IS A REMARK
RESTORE	Resets data statement	900 RESTORE
STOP	Stops program execution	900 STOP
WAIT	Terminal handshake pause	900 WAIT 0, 128, 128

7.0 MSA BASIC FUNCTIONS

FUNCTION	DESCRIPTION	EXAMPLE
ABS	Absolute value	100X=ABS(Y)
ASC	ASCII conversion	100 X=ASC($)
ATN	Arctangent (Radians)	100 X=ATN (¢)
CHR$	ASCII conversion	100 X$=CHR$(7)
COS	Cosine (Radians)	100 S-COS(Y)
EXP	Natural power to e	100 X=EXP(Y)
FRE	Free memory for program	100 PRINT FRE (X)
INP	Input	100 S-INP(1)
INT	Integer	100 X=INT(Y)
LEFT$	Left string-substring	100 X$= LEFT$(Y$,5)
LEN	String length	100 X=LEN(X)
LOG	Natural log	100 X=LOG (Y)
MID$	Substring	100 X$=MID$ (Y$,5,2)
RND	Random number	100 X=INT (RND(5) *6)+1
POS	Print position	100X=POS(X)
RIGHT$	Right string-substring	100X$=RIGHT$ (Y$,5)
SGN	Sign	100 X=SGN(Y)
SIN	Sine (Radians)	100 X=SIN(Y)
SPACE$	Space string	100 X$=SPACE$(5)
SPC	Space print	100 PRINT SPC (25)
SQR	Square root	100 S-SQR(Y)
STR$	Number to string	100 X$=STR$(X)
TAB	Print tab	100 PRINT TAB (25)
TAN	Tangent (Radians)	100 X=TAN(Y)
USR	User defined function	100 USR(X)
VAL	String to number	100 X=VAL(X$)

8.0 SPECIAL CHARACTERS

CHARACTER	DESCRIPTION
@	Erases current line
:	Multiple statement terminator
?	Print
C	Stops program execution
U	Same as @
RUB	Deletes previous character
DEL	Same as RUB
←	Same as RUB
SPACE	Stops program execution temporarily until another character is typed
OR ↑	Raise a number to a power. EXAMPLE: X↑3 computes the third power of X

NOTE: C and U imply control characters.

9.0 ERROR MESSAGES

ERROR MNEUMONIC	ERROR DESCRIPTION
BS	Bad subscript
CN	Can't continue
DD	Double dimension
FC	Function call
ID	Illegal direct
LS	Too long string
NF	Next without for
OD	Out of data
OM	Out of memory
OS	Out of string space
OV	Overflow
RG	Return without GOSUB
SN	Syntax error
ST	String error
TM	Type mismatch
UF	Undefined user function
UL	Unlisted line number
/0	Divide by zero

ADDENDUM 1

Note: This addendum applies only to users who are using CUTER systems to run MSA BASIC.

If you are using CUTER to run MSA BASIC there is a check that should be made before you run MSA BASIC. Your input device should have bit 8 tied to ground or a logical "0". If this is not the case, then either modify the hardware by jumpering bit 8 to ground or make the software modifications listed below.

SOFTWARE MODIFICATION FOR CUTER SYSTEMS

```
GET BASIC
EN 55C
:FE 7F/
SAVE BASIC 0 1FFF
EX 1C00
```

This modification will load BASIC then alter two bytes in the program and save the modified version back on cassette tape. MSA BASIC may then be started at 1C00. Future loads may use the XEQ BASIC format with no alteration.

We hope this modification will not cause any problems in using MSA BASIC and please note that it will apply to only a few of the systems that will be using MSA BASIC.

CORRECTIONS

Referencing Page 9, Special Characters; raise a number to a power should be ^ OR ↑.

ADDENDUM 2

There has been some confusion about saving and loading program cassette tapes with MSA BASIC. We hope the following addendum will help clear up some of the misconceptions.

TO SAVE A PROGRAM ON CASSETTE TAPE

After your program has been typed into the computer, type:

SAVE "X" (Note that the separating space is necessary.)

"X" may be any one letter file name.

TO LOAD A PROGRAM FROM CASSETTE TAPE

Type:

LOAD "X" (Note that the separating space is necessary.)

"X" is the name of the program that was saved using the procedures from above.

TO VERIFY THAT A PROGRAM HAS BEEN SAVED

Type:

LOAD ?"X" (Note that the separating space is necessary.)

"X" is the name of the program that was saved. This procedure will not alter the contents of the program buffer. The computer should respond with:

READY

to verify a good SAVE. If it does not, then you should return to the monitor operating system (SOLOS or CUTER) and execute at 0. Then the procedures to SAVE a program should be repeated.